MATH UN3025 - Midterm 2 Solutions

1. Suppose that $n = p \cdot q$ is the product of 2 prime numbers, p and q. Assume that y is a square mod n and that $y \not\equiv 0 \pmod{n}$.
(a) (5 pts.) How many square roots does y have? Explain your answer.
Solution: Since y is a square mod n, it is also a square mod p and mod q. Every square mod p or q has two square roots $\pm x$ unless it is 0 mod that prime. By the Chinese Remainder Theorem we can combine the square roots mod p and mod q in any way, giving 4 solutions if $y \not\equiv 0 \pmod{p}$ and $y \not\equiv 0 \pmod{q}$, but only 2 solutions if $\gcd(y, n) \neq 1$ (so that y has only one square root modulo p or modulo q). Since $y \not\equiv 0 \pmod{n}$, we must have at least 2 square roots.
(b) (5 pts.) Suppose that you know all of the square roots of y. Explain why you can use this information to factor n.
Solution: From the above, either $\gcd(y, n) \neq 1$ (in which case we get a factor of n from the gcd since $y \not\equiv 0 \pmod{n}$), or y has 4 square roots, call them $\pm a$ and $\pm b$ mod n. Then looking at the possibilities for a and b mod p and q, we must have $a \equiv b \pmod{p}$ and $a \not\equiv b \pmod{q}$ or the same thing with p and q switched, else b would be equal to either a or $-a$. Then $\gcd(a - b, n)$ gives a factor of n.
2. Answer the following two questions about hash functions.
(a) (6 pts.) State each of the 3 desired properties of hash functions.
Solution: These are

1. Easy to compute: given m, there is an efficient algorithm to calculate $h(m)$.
2. Preimage resistant: given y, it is computationally difficult to find m so that $h(m) = y$.
3. Strongly collision free: it is computationally difficult to find m_1, m_2 so that $h(m_1) = h(m_2)$.

(b) (4 pts.) Consider the following function. Given a message m, divide m into blocks of length 160 bits: $m = M_1||M_2||\dots||M_\ell$. Let $h(m) = M_1 \oplus \cdots \oplus M_\ell$, where $\oplus$ is the bitwise XOR function. Which of the three properties of a hash function does h satisfy? (Briefly explain why.)
Solution: Only easy to compute. Any y is its own preimage (by padding it on the left by 0's so that y is 160 bits), so it is not preimage resistant. Also you can take m and pad it with an extra 160 0's to get a collision.
3. The ElGamal signature scheme signing algorithm is as follows. Alice has fixed a public prime p and primitive root α mod p, as well as a secret integer a with $1 \leq a \leq p - 2$. She makes $\beta = \alpha^a \pmod{p}$ public. To sign a message m, she:

1. Selects a secret random k such that $\gcd(k, p - 1) = 1$.
2. Computes $r \equiv \alpha^k \pmod{p}$, where $0 < r < p$.
3. Computes $s \equiv k^{-1}(m - ar) \pmod{p - 1}$.

The signed message is the triple (m, r, s).

(a) (4 pts.) Fill in the blanks in Bob's verification algorithm. (You don't need to prove that it works.)

1. Compute $v_1 \equiv \beta^r r^s \pmod{p}$ and $v_2 \equiv \alpha^m \pmod{p}$.
2. Check whether $v_1 \equiv v_2 \pmod{p}$. If so, declare that the signature is valid.

(b) (4 pts.) Suppose u, v are any numbers such that $\gcd(v, p-1) = 1$. Compute $r = \beta^v \alpha^u \pmod p$ and $s \equiv -rv^{-1} \pmod{p-1}$. Prove that (r, s) is a valid signature for $m = su \pmod{p-1}$. (This is the existential forgery attack from your homework.)
Solution: We have $\beta^r r^s \equiv \alpha^{ar+sav+su} \equiv \alpha^{ar-ar+m} \equiv \alpha^m \pmod p$.
(c) (2 pts.) Explain how hash functions can be used in order to prevent the preceding attack. More precisely, if sign_A denotes Alice's signing function and h is a hash function, explain why it is hard to use the existential forgery attack to construct a triple of values $(m, h(m), \text{sign}_A(h(m)))$.
Solution: The existential forgery attack produces pairs $(y, \text{sign}_A(y))$ with ease, but with no control over what y looks like. So to provide a triple $(m, h(m), \text{sign}_A(h(m)))$ using this attack, we would need to find an m with $h(m) = y$, which is hard by preimage resistance.
4. (5 pts.) Consider the following protocol. Let p be a large prime and α a primitive root. Let a be an integer and let $\beta = \alpha^a \pmod p$. Suppose p, α, and β is public, and that Peggy wants to prove to Victor that she knows a without revealing it. They agree to use the following protocol.

1. Peggy chooses a random number $r \pmod{p-1}$.
2. Peggy computes $h_1 \equiv \alpha^r \pmod p$ and $h_2 \equiv \alpha^{a-r} \pmod p$ and sends them to Victor.
3. Victor chooses $i = 1$ or $i = 2$ and asks Peggy to send either $r_1 = r$ or $r_2 = a - r \pmod{p-1}$.
4. Victor verifies that $h_1 h_2 \equiv \beta$ and that $h_i \equiv \alpha^{r_i} \pmod p$.

They repeat this several times.

Suppose that Eve is trying to pretend to be Alice by claiming to Victor that she knows a. Assume that Eve has a guess for Victor's choice of i. In terms of Eve's guess (either 1 or 2), what values of h_1 and h_2 should Eve send Victor in each round? (Your choices of h_1 and h_2 should be such that if Eve's guess is right, she is able to respond to Victor's challenge.)
Solution: Eve generates $r \pmod{p-1}$ as above. The given values $h_1 \equiv \alpha^r$, $h_2 \equiv \alpha^{a-r}$ work for $i = 1$, but she computes h_2 via $h_2 \equiv \beta\alpha^{-r} \pmod p$. If $i = 2$, she sets $h_2 \equiv \alpha^r$ and defines $h_1 \equiv \beta\alpha^{-r}$ instead. In both cases, she responds to Victor's challenge with r if her guess is correct.
5. Let $\mathbb{F}_2$ be the finite field of 2 elements, and let $P(X) = X^4 + X^3 + 1$ in $\mathbb{F}_2[X]$.
(a) (3 pts.) Prove that $P(X)$ is irreducible. You may assume that $X^2 + X + 1$ is the only irreducible polynomial of degree 2 in $\mathbb{F}_2[X]$.
Solution: If not, $P(X) = Q_1(X)Q_2(X)$ for lower degree polynomials Q_1, Q_2. If either one has degree 1, then P has a root in $\mathbb{F}_2$, but $P(0) = P(1) = 1$, so this is not the case. So Q_1 and Q_2 have degree 2. We can assume both are irreducible, else again P would have a linear factor and thus a root. So $P(X)$ must be $(X^2 + X + 1)^2 = X^4 + X^2 + 1$, but this is not the case. So $P(X)$ is irreducible.
(b) (7 pts.) Define a finite field of 16 elements by $\mathbb{F}_2[X] \pmod{P(X)}$. Find the inverse of $X^2 + 1$ in this field.
Solution: We employ the division algorithm to get $X^4 + X^3 + 1 = (X^2 + X + 1)(X^2 + 1) + X$ and $X^2 + 1 = X \cdot X + 1$. We then use the Extended Euclidean algorithm: we calculate $x_0 = 0, x_1 = 1, x_2 = -(X^2 + X + 1) \cdot 1 + 0 = X^2 + X + 1$, $x_3 = -X(X^2 + X + 1) + 1 = X^3 + X^2 + X + 1$. The algorithm gives the formula $(X^2 + 1)x_3 + (X^4 + X^3 + 1)y_3 = 1$, which implies that $X^3 + X^2 + X + 1$ is the inverse. (Note that to determine the inverse, we did not need to know the actual value of y_3.)
6. (a) (5 pts.) Let E be the elliptic curve $y^2 \equiv x^3 + 2x + 1 \pmod 3$. Find all the points on E.
Solution: We try $x = 0, 1, 2$ and find that $x^3 + 2x + 1 \equiv 1 \pmod 3$ each time. Since 1 has 1 and 2 as square roots modulo 3, we find $(0, 1), (0, 2), (1, 1), (1, 2), (2, 1), (2, 2)$, and ∞ for our list of points.
(b) (5 pts.) How many points P on E satisfy $P + P = \infty$?
Solution: From the doubling formula we see that for $P = (x, y)$, $2P = \infty$ is the same as $y = 0$, which is never the case for our E. This leaves $P = \infty$ as the only solution.
(c) (5 pts.) Find a point Q on E that satisfies $(1, 1) + Q = (1, 2)$.

Solution: Add $-(1,1) = (1,2)$ to both sides to get $Q = 2(1,2)$; the doubling formula shows that $Q = (2,2)$.

Extra credit. (a) (1 pt.) Suppose that two sets of r objects are drawn from the same set of size N. What is the formula from class giving an approximate probability of a match between an object in the first set and an object in the second?

Solution: $1 - e^{-\lambda}$, where $\lambda = r^2/N$.

(b) (4 pts.) Suppose that h is a hash function mapping to strings of $n = 60$ bits. Explain in a few sentences the method, discussed in class, that would allow Fred the Forger to trick Alice into signing the hash of a legitimate contract C, while simultaneously obtaining her signature on the hash of a fraudulent contract F. Assume that a success probability of $1 - \frac{1}{e}$ is acceptable.

Solution: Fred finds 30 spots in both the legitimate and fraudulent contracts where an unmeaningful change can be made. Then he hashes all the 2^{30} possible versions of both contracts that can be made by making or not making each of these changes. The probability that $h(C) = h(F)$ for some versions C and F of the legitimate and fraudulent contracts is $1 - \frac{1}{e}$ by the preceding formula. Then Fred asks Alice to sign the hash of C, which is also the hash of F.

(c) (1 pt.) How can Alice avoid falling for such a trap?

Solution: Alice makes her own unmeaningful change before signing C; to find a hash of a fraudulent contract that matches that of C would require trying around 2^{60} rather than 2^{30} versions, which is too many.